# 

**Linear Regression Transformation & Interpretation - Stata**

Today we’re talking about how to interpret linear regression models, and all you need to know to transform variables.

# 

# Linear Regression Transformation & Interpretation

## Assumptions for Linear Regression

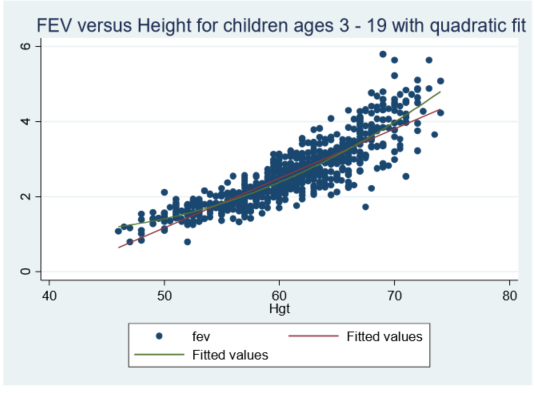
We will consider 4 assumptions to determine if multiple linear regression is a good fit for the structure of our data. Diagnostic analyses are used to determine whether or not these assumptions are valid:

1. Linearity
2. Normality
3. Variance of Y does not depend on the level of X (homoscedasticity)
4. Independent Observations

After you have considered all of your assumptions, you may determine that a linear regression model is not a good fit, and you need to transform a variable.

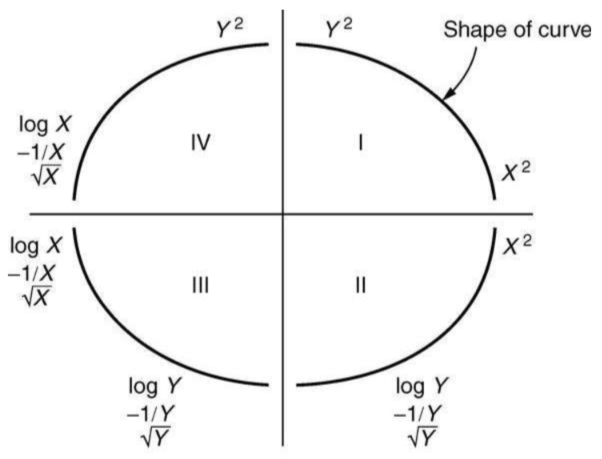
## Determining When a Transformation is Needed

The goal of transformation is to select the variables that provide the best linear regression model to describe the relationship between the dependent (outcome) and independent variables (predictors or factors).

Identifying the appropriate form for each variable is important. Take a look at this scatter plot of forced expiratory volume (FEV) and height. FEV is a common marker for respiratory health.

The red line represents a linear line, and the green line represents a quadratic fit of the height variable (height\*height). However, when we re-assess the assumptions for this model with a quadratic height variable added, we don’t meet the assumptions for linear regression.

## Choosing a Transformation

So how do we choose the appropriate transformation? Use this figure to determine which log transformation is most appropriate given the curve of the fitted line in your scatter plot.

In Epidemiology, a common transformation is the log transformation of the outcome variable. In our previous example, we see that this is an appropriate transformation given the curve of our fitted line. What we aim to do with log transformation is bring in those more extreme values in our data and make them less extreme.

Remember that for every transformation you need to re-assess the assumptions of the model before moving to interpretation.

## Interpretation of Log Transformation

For variables that are measured on a normal scale in linear regression, the coefficients in the model are the **change in the average value of the outcome variable for a 1 unit increase in the predictor variable.**

### In Stata - Log Predictor

When we log transform the **predictor** variable, we calculate:

𝛽\*log(x)

For x we need to specify the percent change. In this equation we multiply our coefficients by a certain percentage - e.g. 𝛽\*log(1.01) for a 1% increase, or 𝛽\*log(1.05) for a 5% increase.

We interpret this as **the change in the average value of the outcome variable for a percent increase in the predictor.**

For example, from our model we get a coefficient for log transformed height of 2.5. Using our equation in Stata, use the following code:

display 2.5\*log(1.01)

You will get **0.0248**.  
*Interpretation:* A 1% increase in height leads to a 0.025 increase in FEV.

In Stata - Log Outcome

When we log transform the **outcome** variable, we calculate:

100\*(e𝛽 - 1)

We interpret this as **the percentage increase in the average value of the outcome variable for a unit increase in the predictor.**

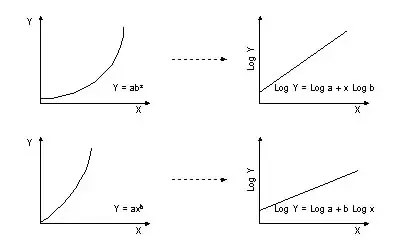
For example, from our model we get a coefficient for Age of 0.025. Using our equation in Stata use the following code:

display 100\*(exp(0.025) - 1)

You will get **2.53**.   
*Interpretation*: A one-year increase in age leads to a 2.5% increase in FEV.

## **Quick Review**

Remember, your assumptions need to be met in order to make good inferences about your data. Influential observations or outliers in your data can significantly impact the interpretation of your results. Think critically about how you will interpret transformed variable results and how meaningful they are when considering transformations.



Twoway (scatter logdrivers predictor)

Two options, choose 1

Generate logoutcome = log(outcome)

Generate predictor2 = predictor\*predictor

Example: generate driver2 = driver\*driver

regress logoutcome predictor